Statistics Canada Statistique Canada

THE DISTRIBUTION OF THE FREQUENCY OF OCCURRENCE OF NUCLEOTIDE SUBSEQUENCES BASED ON THEIR OVERLAP CAPABILITY

by

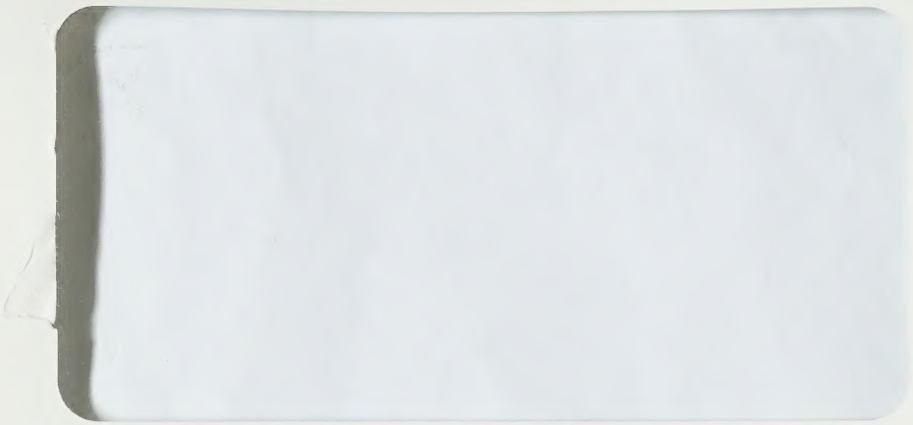
Jane F. Gentleman and Ronald C. Mullin

No. 14

Statistics Canada
Analytical Studies Branch

Research Paper Series

Canada



THE DISTRIBUTION OF THE FREQUENCY OF OCCURRENCE OF NUCLEOTIDE SUBSEQUENCES BASED ON THEIR OVERLAP CAPABILITY

by

Jane F. Gentleman and Ronald C. Mullin

No. 14

Social and Economic Studies Division
Statistics Canada
1988

The Distribution of the Frequency of Occurrence of Nucleotide Subsequences, Based on Their Overlap Capability

Jane F. Gentleman
Statistics Canada, Social and Economic Studies Division
Ottawa, Ontario, Canada K1A 0T6

Ronald C. Mullin
University of Waterloo, Dept. of Combinatorics and Optimization
Waterloo, Ontario, Canada N2L 3G1

ABSTRACT

DNA's genetic code can be represented as an alphabetic sequence composed of the four letters A, C, G, and T, which represent the four types of nucleotides - adenylic, cytidylic, guanylic, and thymidylic acid - of which DNA is composed. Now that these sequences have been identified for many genes and are available in computer-readable form, scientists can analyze these data and search for patterns in an attempt to learn more about the regulatory functions of the gene. One area of study is that of the frequency of occurrence of specific nucleotide subsequences (e.g., ACAC) within part or all of a nucleotide sequence. This paper derives the probability distribution of the frequency of occurrence of a subsequence within a nucleotide sequence, under the hypothesis that the four nucleotides occur at random and with equal probability. This distribution is nontrivial because different subsequences have different "overlap capability." For example, the subsequence AAAA can occur up to 17 times in a sequence of length 20 (which would happen if the sequence were composed solely of A's), but the subsequence ACGT cannot occur more than 5 times in a sequence of length 20. Thus, the frequency distributions are different for each type of overlap capability. It is of interest to assess and compare the degree of nonrandomness for different subsequences or among different portions of a sequence; the existence and degree of nonrandomness may be related to the type and degree of functionality of a nucleotide (sub)sequence. Using the frequency distributions provided here, exact significance tests of the hypothesis of randomness can be performed. An approximate test is also described for use with long sequences; this can be used to test a more general null hypothesis of nucleotides occurring with unequal probabilities.

1. INTRODUCTION.

Genes are long, double-stranded, helical molecules of DNA. Each of the two strands contains a sequence of nucleotides - typically between 1,500 and 15,000 of them - and the two strands are loosely bound together by hydrogen bonds. A

Key words: DNA subsequences; nucleotide subsequences; overlap capability

nucleotide in DNA is identified according to which of four nitrogenous bases it contains: a purine base, adenine (A) or guanine (G), or a pyrimidine base, cytosine (C) or thymine (T). There is a one-to-one correspondence between nucleotides on opposite strands; an A, G, C, or T on one strand is weakly bonded to its complement, T, C, G, or A, respectively, on the other. DNA's so-called "genetic code" can thus be represented as a single alphabetic sequence composed of these four letters. It is by means of this code that the gene controls the formation of other substances in the cell (see, e.g., Guyton (1969)). Also, certain sequences of nucleotides form oncogenic genes which can initiate some forms of cancer.

Relatively recent advances in biochemistry have made it possible to determine the nucleotide sequences for large numbers of genes, and for intergenic material, which also consists of nucleotide sequences. Such data are now available in computer-readable form, so it is possible to look for and analyze patterns within sequences using statistical and statistical computing techniques. Scientists are now able to use pattern recognition algorithms to "learn more about the regulatory nature of the various genetic functional domains, and more about what it is that is recognized within those domains by the cellular hardware..." (Sadler, Waterman, and Smith (1983)). Weir (1985) provides a useful survey of new problems of statistical analysis which have arisen following advances in molecular genetics.

One area of study is that of the frequency of occurrence of specific short nucleotide subsequences (e.g., ACAC) within part or all of a nucleotide sequence. Maizel et al. (1981) concluded that, among the computer programs being widely used for nucleic acid analysis, "Most frequently used are programs that search for occurrences of short subsequences that are used by enzymes as signals to recognize, modify, and express nucleic acids, that determine the frequency and locations of short strings of nucleotides, and that translate nucleic acid sequences into amino acid sequences or complementary polynucleotide strands." Some examples of papers which deal with the analysis of subsequence frequencies are: Aquadro and Greenberg (1983), Gentleman et al. (1984), Grantham et al. (1981), Harr, Haggstrom, and Gustafsson (1983), Korn and Queen (1984), Nussinov (1984), Sadler et al. (1983), Smith and Burks (1983), Smith, Waterman, and Sadler (1983), Queen and Korn (1980), and Vass and Wilson (1984).

It is of interest to assess and compare the degree of nonrandomness for different subsequences or among different portions of a sequence; the existence and degree of nonrandomness may be related to the type and degree of func-

tionality of a nucleotide (sub)sequence. Orgel and Crick (1980) classified the DNA of higher organisms as falling into two classes, one specific and the other comparatively nonspecific. Regarding the latter, they noted that "there is a large amount of evidence which suggests, but does not prove, that much DNA in higher organisms is little better than junk." Vass and Wilson (1984) describe statistical tests for detecting nonrandom arrangements on a nucleotide strand. These tests "may be especially useful in the analysis of patterns in DNA sequences which may partly reflect the structure and function of the genes in which they are part." Shukla and Srivastava (1985) developed a test for sequence randomness based on the frequency of occurrence of a particular subsequence at two positions which are a fixed number of bases apart. They reasoned that "low probability of chance occurrence calls for further exloration...of...possible structural or functional significance, ...(whereas, if there is) a high probability of chance occurrence, then one has to exercise some caution before attaching any structural or functional role to that kind of...repeat."

Gentleman et al. (1984) gave two examples of how the occurrence of a subsequence in unexpectedly large numbers may provide information about structure or function: (1) This phenomenon may reflect the fact that that segment of DNA was originally formed by replication of smaller segments. Such replication might be exact initially, but might be altered with time. But in regions of the sequence where retention of function is necessary, exact repeats would be expected to occur. (2) A break in DNA usually occurs at different positions, from 3 to 10 nucleotides apart, on the two strands. When this happens, a gap is created on each strand opposite remaining nucleotides on the other strand. On each strand, to fill this gap, nucleotides complementary to their counterparts on the other strand are added, thus duplicating the subsequence on the other side of the break. The break itself is filled in. The occurrence of repeated subsequences may therefore identify locations where an insert has been introduced into the DNA sequence.

Relatively little is know about the specific functionality of DNA. The attempt to identify of nucleotide (sub)sequences of greater or lesser randomness is based in part on the concept that a higher degree of functionality may be indicated by a lower degree of randomness. The complementarity of the two strands of DNA permits each strand to act as a template on cell division to form two identical double stranded structures. As DNA reproduces itself, chance mutations may occur, so that DNA is subject to the forces of natural selection and evolution. Thus, a particular configuration in DNA that exists now may have

been favored by natural selection. (For further discussion of genetic evolution, see Doolittle and Sapienza (1980), Orgel and Crick (1980), and Forbes and Shadbolt-Forbes (1988).)

This paper derives the probability distribution of the frequency of occurrence of a subsequence within a nucleotide sequence, under the hypothesis that the four nucleotides occur at random and with equal probability. A general algorithm is provided for calculating this probability distribution, which depends on the sequence length, the subsequence length, and a property of the subsequence which will be termed "overlap capability". Explicit formulas are given for all subsequences of length 2-8. Access to these distributions permits the use of exact significance tests of the hypothesis of randomness. An approximate test is also provided for use when the sequence is long. The observed significance level of such tests measures the extent of the data's departure from the hypothesis, i.e., the degree of nonrandomness.

A null hypothesis of equiprobable occurrence of the different nucleotides is reasonable in the context of the present DNA structures having evolved from a "primordial soup" or "base pool" containing equal quantities of each base. This is discussed by Sege and Saxberg (1982), who provide a statistical test for the simultaneous comparison of several nucleotide subsequences. Their "null hypothesis which one seeks to reject" is that the observed data came by chance selection from a base pool with specified relative frequencies of A, C, G, and T. They describe three alternatives for choosing the four null probabilities:

> "(1) The abstract nucleotide pool is unlimited and therefore the distribution of nucleotides is effectively equal; (2) the sequences are drawn from a pool comprised of the nucleotide distribution typical of the species; or (3) the nucleotide pool for the class of sequences examined is well represented by the total distribution of nucleotides in the sequences themselves."

Sege and Saxberg then discuss the conditions under which each choice is appropriate:

> "The virtual pool selected will be a function of the question posed by the experimenter and the level of information desired. Clearly the most readily interpreted virtual pools are the even virtual pool (frequencies = 0.25) and the 'species/organism' virtual pool (average frequencies of bases for the species/organism). These virtual pools should serve as standards unless the investigator has sufficient reason to warrant

> another type (e.g., experimental virtual pool). When a non-standard virtual pool is used, its justification and the meaning of the resulting (significance levels) must be carefully considered."

The exact distributions provided in this paper can be used to test the first of Sege and Saxberg's alternatives, and the approximate test can be used to test any of the three alternatives.

A model in which the four types of nucleotides occur independently has been assumed by some researchers (e.g., those cited in Biggins and Cannings (1987, p. 521)), and hypothesized by others. In the latter case, the hypothesis has been accepted in numerous situations, particularly in the analysis of relatively short (sub)sequences. Garden (1980) fitted Markov chain models to three DNA/RNA sequences, finding that Markov models of order three, two, and zero fitted best. (In RNA, the pyrimidine uracil appears instead of thymine.) The zeroth-order model fitted a gene of length 1632. Fuchs (1980) speculated that the length of the sequence is directly related to the order, citing Garden's further results for subsequences to support this. Fuchs noted that "the majority of the 500-nucleotide segments were fitted well by a model of order zero or one, as expected for short sequences." He recommended two types of supplementary analyses: detection of anomalous regions in a sequence, and analysis of deviations between the observed and expected frequencies of nucleotide subsequences.

Section 2 below defines the concept of "overlap capability", a property of a subsequence which complicates the probability function for the frequency of occurrence of the subsequence. Section 3 derives the expectation and variance of this random variable. The probability function - which is different for each type of overlap capability - is derived in Section 4 (with further details in the Appendix). Section 5 provides examples, using a human genome sequence, of the use of the probability function in exact and approximate significance tests of randomness.

2. DEFINITION OF OVERLAP CAPABILITY.

Assume that the four nucleotides which make up a subsequence occur independently and with equal probability, so that the probability p of the occurrence of a subsequence of length L is $(1/4)^L$. Let the random variable X be the frequency of occurrence of a nucleotide subsequence of length L within a nucleotide sequence of length M. A subsequence "occurs at position i" if it is found to begin at position i. Then $n = M-L+1$ is the maximum value achievable by X. Let $f(x;L,M,Q)$ be the probability function of X. This depends not only on

the scalars L and M, but also on the vector Q, which represents the "overlap capability" of the specific subsequence. As a simple example, the subsequence AAAA can occur between 0 and 17 times within a sequence of length 20. The subsequence ACAC cannot occur more than 9 times in a sequence of length 20 because it has less overlap capability. The subsequence ACGT has no overlap capability except in the trivial case when it is superimposed on itself, so it cannot occur more than 5 times. Thus, f(x;L,M,Q) cannot be treated as a binomial distribution involving independent trials.

Define overlap capability as follows: Let S be a given subsequence and S_1, S_2,...,S_L be letters representing its L nucleotides from left to right. Then represent the overlap capability Q of S as a binary sequence Q_1, Q_2,...,Q_L such that Q_i=1 if it is possible for the subsequence's first i letters to overlap its last i letters, and Q_i=0 otherwise (for i=1,...,L). Specifically, Q_i=1 if $S_k=S_{k+L-i}$ for k=1,...,i, and Q_i=0 otherwise (for i=1,...,L). Obviously, Q_L=1 because a subsequence can always overlap its entire self. For example, the subsequence ACAC has overlap capability 0,1,0,1, and the subsequence AAAC has overlap capability 0,0,0,1. Clearly, many subsequences can have the same overlap capability. On the other hand, not all 2^L possible binary sequences of length L yield possible Q's, due to interrelationships among the elements of Q; for example, no subsequence can have overlap capability 1,0,1,1 (because Q_3=1 implies that all elements of S are the same, but Q_2=0 implies the existence of some inequalities among them). For L=2 to L=8, for example, there are, respectively, only 2, 3, 4, 6, 8, 10, and 13 possible overlap capabilities. An algorithm that can be used to test a binary sequence to determine if it is a possible overlap capability is given in Guibas and Odlyzko (1981). An algorithm to generate all possible overlap capabilties given L is described in Gentleman and Mullin (1987).

The above model can be described in the terminology of Markov chains. Feller (1950, p. 376, Problem 1) descibed a special case of this situation (for L=2 and a two-letter alphabet) as a four-state, first-order Markov process. That approach generalizes here to an a^L-state, (L-1)-order Markov process (where a is the number of letters in the genetic alphabet). Then in particular, the transition probability of the occurrence of S, given that S occurred L-k positions before, is $p^{L-k}Q_k$, for k=1,...,L-1 (where $p=(1/a)^L$). (This is easily further generalized to the case of letters having unequal probabilities.)

Overlap capability enters into the discussion by Biggins and Cannings (1987) of restriction enzymes which cut DNA sequences whenever certain specific

subsequences occur. If one of these subsequences overlaps itself or another of the recognizable subsequences, the cut occurs only at the site of the "earlier" occurrence. For other examples of analyses incorporating the concept of overlap capability, see Shukla and Srivastava (1985) and Karlin and Ost (1987).

3. DERIVATION OF EXPECTATION AND VARIANCE

The expectation of X is, perhaps counterintuitively, independent of the subsequence's overlap capability (Q). The variance depends on Q and, as would be expected, is larger for subsequences with a "higher degree" of overlap capability. For example, if the sequence length M=20 and the subsequence length L=4, then for the subsequences AAAA, ACAC, and ACGT, E(X)=.07 and V(X) is .11, .07, and .06, respectively.

To derive the expectation and variance of X, assume that $M \geq L$, and define indicator variables $Y_1, Y_2, \ldots, Y_n$ (where n=M-L+1) such that $Y_i=1$ if the subsequence S occurs at position i, and $Y_i=0$ otherwise (for i=1,...,n).

Then $X = \sum_{i=1}^{n} Y_i$, so that

$E(X) = \sum_{i=1}^{n} E(Y_i)$. Since $E(Y_i) = (\frac{1}{4})^L = p$, $E(X) = np$, independent of Q, and

$$V(X) = \sum_{i=1}^{n}\sum_{j=1}^{n} \mathrm{Cov}(Y_i,Y_j) = \sum_{i=1}^{n}\sum_{j=1}^{n} E(Y_iY_j) - \sum_{i=1}^{n} E(Y_i) \sum_{j=1}^{n} E(Y_j)$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{n} E(Y_iY_j) - n^2p^2 \qquad (1).$$

To obtain V(X), it is necessary to take account of the fact that the Y's are not in general independent of each other; covariances between Y's which are "near neighbors" depend on Q. To determine these covariances, quantities of the form $E(Y_iY_{i+k})$ must be calculated. This is just the probability that the subsequence occurs at both position i and position i+k. If $0 \leq k \leq \min(L-1,n-1)$, then $E(Y_iY_{i+k}) = Q_{L-k}(1/4)^{L+k} = pQ_{L-k}(1/4)^k$.

If $\min(L-1,n-1) < k < n$, Q is irrelevant and $E(Y_iY_{i+k}) = (1/4)^{2L} = p^2$.

There are n terms among the n^2 in $\sum_{i=1}^{n}\sum_{j=1}^{n} E(Y_iY_j)$ such that i=j.

Also, if $n > 1$ there are 2(n-k) terms such that j=i+k or i=j+k (for k=1,...,min(L-1,n-1)). If $n > L$, there are $2\sum_{k=1}^{n-L} k = (n-L)(n-L+1)$ remaining terms which do not depend on Q. Thus,

$$\sum_{i=1}^{n}\sum_{j=1}^{n} E(Y_i Y_j) = np + p^2(n-L)(n-L+1) + 2p\sum_{k=1}^{\min(L-1,\,n-1)} (n-k)Q_{L-k}(1/4)^k,$$

so Eqn. (1) becomes

$$V(X) = np(1-np) + p^2(n-L)(n-L+1) + 2p\sum_{k=1}^{\min(L-1,\,n-1)} (n-k)Q_{L-k}(1/4)^k \qquad (2).$$

On the right hand side of Eqn. (2) and of the previous equation, the second term equals zero if $n \leq L$, and the third term equals zero if $n=1$. If $L=1$, the third term in Eqn. (2) equals zero, and $V(X)$ reduces, as it should, to the variance $np(1-p)$ of a binomial random variable.

The formulas for $E(X)$ and $V(X)$ can be generalized as follows for an arbitrary number a of letters in the alphabet, occurring independently with probabilities $p_1, p_2, \ldots, p_a$ (which sum to 1). Suppose the subsequence S consists of L letters with respective probabilities $p_{j_1}, p_{j_2}, \ldots, p_{j_L}$. Let $P_k = \prod_{m=1}^{k} p_{j_m}$ be the product of the probabilities of the first k letters in S. In particular, $P_L = \prod_{m=1}^{L} p_{j_m} = Pr(S)$. Then $E(X) = nP_L$, and Eqns. (1) and (2) may be generalized by substituting P_L for p, and P_k for $(1/4)^k$.

The contribution of Q to $V(X)$ in Eqn. (2) motivates the following procedure for ranking subsequences in order of their degree of overlap capability: Let $\overline{Q}$ be the binary number constructed from the L elements of Q in reverse order. Then subsequence S_1 with overlap capability Q_1 has greater overlap capability than subsequence S_2 with overlap capability Q_2 if $\overline{Q}_1 > \overline{Q}_2$. Thus, for example, the following subsequences are listed in increasing order of overlap capability: ACGTT, ACGTA, ACGAC, AACAA, ACACA, AAAAA (for which $\overline{Q}$ = 10000, 10001, 10010, 10011, 10101, 11111, respectively). Using this ranking procedure, then for fixed L and $n > 1$, $V(X)$ increases with the degree of overlap capability.

Examination of Eqn. (2) shows that for $n > L$, the maximum value of $V(X)$ (achieved when all elements of Q are equal to 1), is greater than the variance $np(1-p)$ of a binomial random variable, and the minimum $V(X)$ (achieved when all elements of Q except Q_L are equal to 0) is less than $np(1-p)$. As $n \to \infty$, $V(X)-np(1-p) \to 2np \sum_{k=1}^{L-1} (4^{-k}-p)$ if $Q=1,1,\ldots,1$, and $V(X)-np(1-p) \to -2(L-1)np^2$ if $Q=0,0,\ldots,0,1$. Thus, the difference between the maximum and minimum variance approaches $2np \sum_{k=1}^{L-1} 4^{-k}$ as $n \to \infty$. For example, these three limiting quantities are equal, respectively, to .023n, -.00781n, and .0313n for L=2, and to .00266n, -.0000916n, and .00256n for L=4.

4. DERIVATION OF THE PROBABILITY FUNCTION

Using combinatorial theory, an algebraic formula can be derived for the probability generating function of X. This requires an appropriate application, as described in the Appendix, of the combinatorial techniques in Goulden and Jackson (1983, Section 2.8.). From the probability generating function, an algebraic formula for $f(x;L,M,Q)$, recursive in x and M, can be obtained as shown below. A separate formula involving parameters L and M is required for each Q.

The probability generating function $P(u,v)$ for $f(x;L,M,Q)$ is

$$P(u,v) = \sum_{M=0}^{\infty} \sum_{x=0}^{\infty} f(x;L,M,Q) u^M v^x$$

$$= \frac{1-(v-1)h(u/4)}{[1-(v-1)h(u/4)](1-u)-(u/4)^L(v-1)} \qquad (3).$$

This formula involves the "prefix polynomial" $h(x)$, defined as

$h(x) = \sum_{k=1}^{L-1} x^{L-k} Q_k$. (Note that $h(p)$ is the sum of the L-1 transition probabilities described in Section 2.) In Eqn. (3), the "4", which appears three times, can be generalized to be the number of equiprobable letters of the alphabet. This also holds for Eqns. (4), (5), and (8) below.

$f(x;L,M,Q)$ is the coefficient of $u^M v^x$ in Eqn. (3). To obtain a formula for f, write the denominator of the right hand side of Eqn. (3) as 1-D, where

$$D = u - (u/4)^L + (u/4)^L v - (1-u+uv-v)h(u/4) \qquad (4).$$

Then multiply both sides of Eqn. (3) by 1-D and isolate $P(u,v)$ on the left hand side to obtain

$$\sum_{M=0}^{\infty}\sum_{x=0}^{\infty} f(x;L,M,Q)u^M v^x = 1-(v-1)h(u/4) + D\sum_{M=0}^{\infty}\sum_{x=0}^{\infty} f(x;L,M,Q)u^M v^x \qquad (5).$$

If D is written as

$$D = \sum_{i=1}^{L}\sum_{j=0}^{1} C_{ij}u^i v^j$$

so that C_{ij} is the coefficient of $u^i v^j$ in Eqn. (4), then the coefficient of $u^M v^x$ in Eqn. (5) is

$$f(x;L,M,Q) = \sum_{i=1}^{L}\sum_{j=0}^{1} C_{ij}\, f(x-j;L,M-i,Q) \qquad (6).$$

This is obtained by applying the following boundary conditions:

$$\begin{aligned} &f(0;L,M,Q) = 1 \text{ if } M < L; \\ &f(x;L,M,Q) = 0 \text{ if } M < L \text{ and } x > 0 \end{aligned} \qquad (7).$$

The following general formulas for C_{ij}, obtained by expanding and rearranging terms in Eqn. (4), can thus be used in Eqn. (6) to obtain a formula for $f(x;L,M,Q)$:

For $L>1$, define C_{ij} $(i=1,\ldots,L;j=0,1)$ as follows:

$$\left.\begin{aligned} C_{1,0} &= (4-Q_{L-1})/4 \\ C_{1,1} &= Q_{L-1}/4 \\ C_{L,0} &= (4Q_1-1)/4^L \\ C_{L,1} &= (1-4Q_1)/4^L = -C_{L,0} \end{aligned}\right\} \qquad (8).$$

Also, if $L > 2$, then for $k=1$ to $L-2$:

$$C_{L-k,0} = (4Q_{k+1} - Q_k)/4^{L-k}$$

$$C_{L-k,1} = (Q_k - 4Q_{k+1})/4^{L-k} = -C_{L-k,0}$$

Eqn. (6) is applicable for $M=L,\ldots,\infty$ and $x=0,\ldots,M-L+1$. If $x=0$, then terms involving the argument $x-1$ are equal to zero. Table 1 provides formulas for $f(x;L,M,Q)$ for all possible Q's for $L=2$ to $L=8$.

Using the subsequence ACA as an example, so that $L=3$ and $Q=1,0,1$, the C_{ij}'s are obtained as follows:

$$C_{1,0} = 1$$
$$C_{1,1} = 0$$
$$C_{3,0} = 3/4^3$$
$$C_{3,1} = -3/4^3$$
$$C_{2,0} = -1/4^2$$
$$C_{2,1} = 1/4^2$$

Therefore, from Eqn. (6),

$$f(x;L,M,Q) = f(x;L,M-1,Q) - f(x;L,M-2,Q)/16 + 3f(x;L,M-3,Q)/64$$

$$+ f(x-1;L,M-2,Q)/16 - 3f(x-1;L,M-3,Q)/64$$

as in Table 1.

The formulas thus derived for $f(x;L,M,Q)$ are recursive; $f(x;L,M,Q)$ depends in general on $f(x;L,M-i,Q)$ and $f(x-1;L,M-i,Q)$ for $i=1,\ldots,L$. (The proof of this follows from Eqn. (3) and from the fact that no term of the prefix polynomial $h(u/4)$ can be of degree greater than $L-1$.) Thus, a computer program to calculate numeric values of $f(x;L,M,Q)$ for $x = 0$ to an upper limit J needs to store an L by J-L array of probabilities. Alternatively, a recursive programming language such as Pascal or Algol can be used. In either

case, initialization is performed using the boundary conditions of Eqn. (7). A Fortran program to compute f(x;L,M,Q), E(X), and V(X) given L, M, and the subsequence is available from the authors.

Eqn. (6) is valid for the binomial distribution (i.e., for L=1 and Q=1) if $C_{1,0} = \frac{3}{4}$ and $C_{1,1} = \frac{1}{4}$, yielding the recursion formula

$$f(x;L,M,Q) = 3f(x;L,M-1,Q)/4 + f(x-1;L,M-1,Q)/4 \qquad (9).$$

Note from Eqn. (8) that if $L > 1$, the only case in which $C_{1,0} = \frac{3}{4}$ and $C_{1,1} = \frac{1}{4}$ is when $Q_{L-1}=1$, in which case $Q_1=Q_2=\ldots=Q_{L-2}=1$, so that all remaining C_{ij}'s in Eqn. (6) are nonzero.

Table 2 shows values of f(x;4,20,Q) for the three subsequences AAAA, ACAC, and ACGT, chosen to represent subsequences having "high", "medium", and "low" degrees of overlap capability, respectively.

5. EXAMPLES OF THE USE OF THE PROBABILITY FUNCTION IN EXACT AND APPROXIMATE SIGNIFICANCE TESTS.

The availability of formulas for $f(x;L,M,Q)$ makes it possible to perform exact significance tests of the hypothesis of randomness, and the formulas for $E(X)$ and $V(X)$ provide the needed quantities for an approximate test. As an example, an 825-nucleotide-long sequence obtained from Georgetown University Medical Center's Nucleic Sequence Database and shown in Table 3 will be used. It is described in Dayhoff et al. (1983) as a "Middle repetitive (Alu family) genome fragment - human (length 825)." This intergenic material was one of 14 sequences examined in Gentleman et al. (1984).

The subsequence CC occurs 67 times in this genome fragment. Under the hypothesis, the frequency of its occurrence in a sequence of length 825 has an expected value of 51.50 and a variance of 67.57. Table 4 shows probabilities and cumulative probabilities for frequencies from 30 through 70. (The complete range is from 0 through 824.) Using these probabilities, the significance level for an exact test of the hypothesis can be calculated as the sum of the probabilities for frequencies ≥ 67, plus the sum of the probabilities for frequencies ≤ 36 (these being all frequencies with probabilities less than or equal to $Pr(67)$). Thus, the significance level = .039 + .028 = .067.

By identifying frequencies as close as possible to the .025 and .975 points of this discrete distribution, a 95% confidence interval is obtained; its lower bound is between 35 and 36, and its upper bound is between 67 and 68.

The usual approximate Chi-square goodness-of-fit test has sometimes been used to compare observed and expected subsequence frequencies. (For example, Smith et al. (1983) used this test with expected frequencies based on the overall sequence base composition.) The goodness-of-fit test sums the scaled squared differences between observed and expected frequencies. The two observed frequencies in the present example are 67 (the number of occurrences of CC) and 757 (the number of occurrences of other subsequences of length two). The resulting test statistic value is 4.976, so the approximate significance level for this test would normally be calculated as $Pr(\chi^2_1 \geq 4.976) = .0257$, which is considerably smaller than the exact value of .0670. However, the goodness-of-fit test is inappropriate here, due to differences, which remain even as $n \rightarrow \infty$, between $f(x;L,M,Q)$ and the binomial distribution (as shown in Sections 3 and 4); when there are only two observed frequencies, the goodness-of-fit test statistic is equivalent to the square of a standardized observed binomial(n,p) frequency.

An appropriate approximate test statistic can be obtained by standardizing

the observed frequency of a subsequence using the correct variance (as given in Eqn. (2)), i.e., by using $T^* = (x-np)^2/V(X)$ instead of $T = (x-np)^2/[np(1-p)]$. The central limit theorem for dependent trials can then be invoked (e.g., as in Feller (1950), p. 374, and in Shukla and Srivastava (1985)) and a χ^2_1 approximation used for sufficiently large n. In the case of the x=67 occurrences of the subsequence CC, T^*=3.5556, yielding a much more accurate approximate significance level of $Pr(\chi^2_1 \geq 3.5556) = .0594$. (Note that T^* can be used in the more general case of an a-letter alphabet with probabilities that are not necessarily equal.)

Examining a longer subsequence, the observed frequency of the subsequence CCCC is found to be 6. The expectation and variance of the exact distribution are 3.21 and 5.23, respectively. The exact significance level is .153, and the approximate significance level using T^* is .223. (The approximate significance level using T would be .119.) In this case, the approximate test is not accurate; the Chi-square approximation relies on expected frequencies being of about size five or larger, because f(x;L,M,Q) is then more symmetric.

This illustrates the fact that, for fixed n, the approximate test is less likely to be usable for a longer subsequence than for a shorter one, since E(X) decreases as L increases. Fortuitously, computation of an exact significance level is considerably faster (and therefore cheaper) for a larger value of L than for a smaller one; both lower and upper tail areas are required to calculate the P-value for a two-sided test, and when E(X) is relatively small, fewer values of f(x;L,M,Q) need to be calculated recursively before reaching the upper tail of the distribution.

The sequence TTTTTT occurs twice in Table 3. The expected number of occurrences is .20, and the variance is .33. The significance level for the exact test is .042. Since the expected frequency is so small, an approximate test would not be used. (If it were, the resulting significance level would be .00181 for T^* and .00006 for T.)

As a final example, consider the subsequences TTGTTT and AAACAA, which occur six and five times, respectively. These subsequences are inverse complements of each other; each consists of the complementary nucleotides of the other, in reverse order. Each occurs much more often than would be expected; the respective exact significance levels are $.12\times10^{-6}$ and $.31\times10^{-5}$. Perusal of the locations of occurrence of these subsequences reveals that all six occurrences of TTGTTT are close together (beginning in positions 773, 778, 785, 789, and 797), and that four of the five occurrences of AAACAA are close together (in

positions 355, 361, 375, and 387). The two clusters of subsequences occur slightly more than 400 nucleotides apart in the overall sequence. This suggests the possibility that the sequence has a looped superstructure, stabilized by the bonding together of two regions which are about 400 nucleotides apart.

6. CONCLUDING REMARKS.

Formulas have been provided here, and methods described for obtaining any others which are needed, which permit means, variances, and probabilities to be calculated for distributions of nucleotide subsequence frequencies. Exact significance tests can be performed and confidence intervals calculated, or an approximate test can be used, to analyze patterns of nonrandomness in nucleotide sequences or subsequences. This can assist scientists in learning about the structure and functionality of the (sub)sequences. The exact methods are especially useful when the expected subsequence frequency and/or the sequence length is so small that an approximate test is not usable. On the other hand, the approximate test can be used for the more general case where the letters of the genetic alphabet have hypothesized probabilities which are not necessarily equal.

Significance levels from these tests can also be used to compare two or more sequences, as follows: For a given subsequence S, perform the significance test for each sequence and compare the P-values, thus comparing the deviation of the sequences from a common null hypothesis. Comparison of P-values instead of observed frequencies permits sequences of different lengths to be compared, thus avoiding problems of alignment. It also permits results for subsequences having different lengths or overlap capabilities to be compared, since the P-value is standardized according to each subsequence's own frequency distribution.

In analyzing patterns within a sequence, it will be natural to repeat these tests for numerous subsequences, in which case the analyst should bear in mind the usual caveats appropriate for multiple comparisons. One possible approach, in the spirit of Daniel (1959), would be to use a χ^2_1 (or half-normal or normal) probability plot to analyze multiple values of the approximate test statistic T^* (or its square root, or its signed square root, respectively).

When it is appropriate to assess the degree of departure from randomness, these concepts and methods can also be applied to the analysis of sequences of amino acids in proteins, and in fields other than molecular biology, e.g., in time series analysis and cryptography.

It would be useful if future research could produce a generalization of the frequency distribution formulas to the case of unequal probabilities for dif-

ferent alphabet letters.

7. ACKNOWLEDGEMENTS.

The authors wish to thank W.F. Forbes and M.A. Shadbolt-Forbes of the University of Waterloo, whose work motivated the problem addressed in this paper, and who provided much scientific information. We also thank the referees for some very useful suggestions, and Audrey Sirois of Statistics Canada for her careful mathematical typing.

8. REFERENCES.

Aquadro, Charles F. and Greenberg, Barry D. (1983). Human Mitochondrial DNA Variation and Evolution: Analysis of Nucleotide Sequences from Seven Individuals. Genetics, 103, 287-312.

Biggins, J.D. and Cannings, C. (1987). Markov Renewal Processes, Counters and Repeated Sequences in Markov Chains. Adv. Appl. Prob., 19, 521-545.

Daniel, C. (1959). Use of Half-Normal Plots in Interpreting Factorial Two-Level Experiments. Technometrics, 1, 311-341.

Dayhoff, M.O., Chen, H.R., Hunt, L.T., Barker, W.C., Yeh, L.-S., George, D.G., and Orcutt, B.C. (1983). Nucleic Acid Sequence Database, user's manual for June 1983 Release, National Biomedical Research Foundation, Document NASD-0683C, Georgetown Univ. Medical Center.

Doolittle, W. Ford and Sapienza, Carmen (1980). Selfish Genes, the Phenotype Paradigm and Genome Evolution. Nature, 284, 601-603.

Feller, William (1950). An Introduction to Probability Theory and Its Applications. Volume I. Second Edition. Wiley, New York.

Forbes, W.F. and Shadbolt-Forbes, M.A. (1988). The Role of Pattern Analysis of DNA. Presented at ASA Winter Conference on Statistics in Biotechnology, San Antonio.

Fuchs, Camil (1980). On the Distribution of the Nucleotides in Seven Completely Sequenced DNAs. Gene, 10, 371-373.

Garden, Peter W. (1980). Markov Analysis of Viral DNA/RNA Sequences. Journal of Theoretical Biology, 82, 679-684.

Gentleman, Jane F. and Mullin, Ronald C. (1986). The Distribution of the Frequency of Occurrence of Nucleotide Substrings, Based on Their Overlap Capability. COMPSTAT 86, Short communications and posters, Rome, Italy, 97-98.

Gentleman, Jane F. and Mullin, Ronald C. (1987). Algorithms for Calculating the Distribution of the Frequency of Occurrence of Nucleotide Substrings. Unpublished manuscript.

Gentleman, J.F., Shadbolt-Forbes, M.A., Hawkins, J.W., Ladik, J., and Forbes, W.F. (1984). Problems of Pattern Recognition in Nucleotide Sequences. Math. Scientist, 9, 125-139.

Goulden, Ian P. and Jackson, David M. (1983). Combinatorial Enumeration. Wiley Interscience, New York.

Grantham, R., Gautier, C., Gouy, M., Jacobzone, M., and Mercier, R. (1981). Codon Catalog Usage is a Genome Strategy Modulated for Gene Expressivity. Nucleic Acids Research, 9, r43-r74.

Guibus, L.J. and Odlyzko, A.M. (1981). Periods in Strings. Journal of Combinatorial Theory, Series A, 30, 19-42.

Guyton, Arthur C. (1969). Function of the Human Body, Third Edition. W.B. Saunders, Phila., London, Toronto.

Harr, Robert, Haggstrom, Mikael, and Gustafsson, Petter (1983). Search Al-

gorithm for Pattern Match Analysis of Nucleic Acid Sequences. Nucleic Acids Research, 11, 2943-2957.

Karlin, Samuel and Ost, Friedemann (1987). Counts of Long Aligned Word Matches Among Random Letter Sequences. Adv. Appl. Prob., 19, 293-351.

Korn, Laurence Jay and Queen, Cary (1984). Analysis of Biological Sequences on Small Computers. DNA, 3, 421-436.

Maizel, Jacob V., Jr. and Lenk, Robert P. (1981). Enhanced Graphic Matrix Analysis of Nucleic Acid and Protein Sequences. Proceedings of the National Academy of Science USA, 78, 7665-7669.

Nussinov, R. (1984). Doublet frequencies in evolutionary distinct groups. Nucleic Acids Research, 12, 1749-1763.

Orgel, L.E. and Crick, F.H.C. (1980). Selfish DNA: The Ultimate Parasite. Nature, 284, 604-607.

Queen, Cary L. and Korn, Laurence Jay (1980). Computer Analysis of Nucleic Acids and Proteins. Methods in Enzymology, 65, 595-609.

Sadler, J.R., Waterman, M.S., and Smith, T.F. (1983). Regulatory Pattern Identification in Nucleic Acid Sequences. Nucleic Acids Research, 11, 2221-2231.

Sege, Robert D. and Saxberg, Bo E.H. (1982). A Statistical Test for Comparing Several Nucleotide Sequences. Nucleic Acids Research, 10, 375-389.

Shukla, Rakesh and Srivastava, R.C. (1985). The Statistical Analysis of Direct Repeats in Nucleic Acid Sequences. J. Appl. Prob., 22, 15-24.

Smith, Temple F. and Burks, Christian (1983). Searching for Sequence Similarities. Nature, 301, 194.

Smith, T.F., Waterman, M.S., and Sadler, J.R. (1983). Statistical Characterization of Nucleic Acid Sequence Functional Domains. Nucleic Acids Research, 11, 2205-2220.

Vass, J. Keith and Wilson, Richard H. (1984). 'ZSTATS' - A Statistical Analysis for Potential Z-DNA Sequences. Nucleic Acids Research, 12, 825-832.

Weir, B.S. (1985). Statistical Analysis of Molecular Genetic Data. IMA J. Mathematics Applied in Medicine & Biology, 2, 1-39.

TABLE 1: Formulas for f(x;L,M,Q)
for All Possible Q's for L=2 to L=8

For notational brevity, f(x;L,M,Q) is denoted a(M,x).
Formulas are applicable for M=L, and x=0,M-L+1.
If x=0, terms involving the argument x-1 are equal to zero.

L=2:

Q=0,1 (e.g., S=AC)

$$a(M,x) = a(M-1,x) - a(M-2,x)/16 + a(M-2,x-1)/16$$

Q=1,1 (e.g., S=AA)

$$a(M,x) = 3a(M-1,x)/4 + 3a(M-2,x)/16 + a(M-1,x-1)/4 - 3a(M-2,x-1)/16$$

L=3:

Q=0,0,1 (e.g., S=ACC)

$$a(M,x) = a(M-1,x) - a(M-3,x)/64 + a(M-3,x-1)/64$$

Q=1,0,1 (e.g., S=ACA)

$$a(M,x) = a(M-1,x) - a(M-2,x)/16 + 3a(M-3,x)/64 + a(M-2,x-1)/16 - 3a(M-3,x-1)/64$$

Q=1,1,1 (e.g., S=AAA)

$$a(M,x) = 3a(M-1,x)/4 + 3a(M-2,x)/16 + 3a(M-3,x)/64 + a(M-1,x-1)/4 - 3a(M-2,x-1)/16 - 3a(M-3,x-1)/64$$

L=4:

Q=0,0,0,1 (e.g., S=ACGT)

$$a(M,x) = a(M-1,x) - a(M-4,x)/256 + a(M-4,x-1)/256$$

Q=0,1,0,1 (e.g., S=ACAC)

$$a(M,x) = a(M-1,x) - a(M-2,x)/16 + a(M-3,x)/16 - a(M-4,x)/256 + a(M-2,x-1)/16 - a(M-3,x-1)/16 + a(M-4,x-1)/256$$

Q=1,0,0,1 (e.g., S=ACGA)

$$a(M,x) = a(M-1,x) - a(M-3,x)/64 + 3a(M-4,x)/256 + a(M-3,x-1)/64 - 3a(M-4,x-1)/256$$

Q=1,1,1,1 (e.g., S=AAAA)

$$a(M,x) = 3a(M-1,x)/4 + 3a(M-2,x)/16 + 3a(M-3,x)/64 + 3a(M-4,x)/256 + a(M-1,x-1)/4 - 3a(M-2,x-1)/16 - 3a(M-3,x-1)/64 - 3a(M-4,x-1)/256$$

L=5:

Q=0,0,0,0,1 (e.g., S=ACGTT)

$$a(M,x) = a(M-1,x) - a(M-5,x)/1024 + a(M-5,x-1)/1024$$

Q=0,1,0,0,1 (e.g., S=ACGAC)

$$a(M,x) = a(M-1,x) - a(M-3,x)/64 + a(M-4,x)/64 - a(M-5,x)/1024 + a(M-3,x-1)/64 - a(M-4,x-1)/64 + a(M-5,x-1)/1024$$

Q=1,0,0,0,1 (e.g., S=ACGTA)

$$a(M,x) = a(M-1,x) - a(M-4,x)/256 + 3a(M-5,x)/1024 + a(M-4,x-1)/256 - 3a(M-5,x-1)/1024$$

```
Q=1,0,1,0,1 (e.g., S=ACACA)
  a(M,x) = a(M-1,x)
         - a(M-2,x)/16 + a(M-3,x)/16
         - a(M-4,x)/256
         + 3a(M-5,x)/1024
         + a(M-2,x-1)/16 - a(M-3,x-1)/16
         + a(M-4,x-1)/256 - 3a(M-5,x-1)/1024

Q=1,1,0,0,1 (e.g., S=AACAA)
  a(M,x) = a(M-1,x)
         - a(M-3,x)/64
         + 3a(M-4,x)/256 + 3a(M-5,x)/1024
         + a(M-3,x-1)/64
         - 3a(M-4,x-1)/256 - 3a(M-5,x-1)/1024

Q=1,1,1,1,1 (e.g., S=AAAAA)
  a(M,x) = 3a(M-1,x)/4
         + 3a(M-2,x)/16
         + 3a(M-3,x)/64
         + 3a(M-4,x)/256 + 3a(M-5,x)/1024
         + a(M-1,x-1)/4
         - 3a(M-2,x-1)/16
         - 3a(M-3,x-1)/64
         - 3a(M-4,x-1)/256 - 3a(M-5,x-1)/1024

L=6:

Q=0,0,0,0,0,1 (e.g., S=ACGTAG)
  a(M,x) = a(M-1,x) - a(M-6,x)/4096
         + a(M-6,x-1)/4096

Q=0,0,1,0,0,1 (e.g., S=ACGACG)
  a(M,x) = a(M-1,x) - a(M-3,x)/64 + a(M-4,x)/64
         - a(M-6,x)/4096
         + a(M-3,x-1)/64 - a(M-4,x-1)/64
         + a(M-6,x-1)/4096

Q=0,1,0,0,0,1 (e.g., S=ACCCAC)
  a(M,x) = a(M-1,x) - a(M-4,x)/256 + a(M-5,x)/256
         - a(M-6,x)/4096
         + a(M-4,x-1)/256 - a(M-5,x-1)/256
         + a(M-6,x-1)/4096

Q=0,1,0,1,0,1 (e.g., S=ACACAC)
  a(M,x) = a(M-1,x) - a(M-2,x)/16 + a(M-3,x)/16
         - a(M-4,x)/256 + a(M-5,x)/256
         - a(M-6,x)/4096
         + a(M-2,x-1)/16 - a(M-3,x-1)/16
         + a(M-4,x-1)/256 - a(M-5,x-1)/256
         + a(M-6,x-1)/4096

Q=1,0,0,0,0,1 (e.g., S=ACGTCA)
  a(M,x) = a(M-1,x) - a(M-5,x)/1024 + 3a(M-6,x)/4096
         + a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096

Q=1,0,1,0,0,1 (e.g., S=ACAACA)
  a(M,x) = a(M-1,x) - a(M-3,x)/64 + a(M-4,x)/64
         - a(M-5,x)/1024 + 3a(M-6,x)/4096
         + a(M-3,x-1)/64 - a(M-4,x-1)/64
         + a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096

Q=1,1,0,0,0,1 (e.g., S=AAGTAA)
  a(M,x) = a(M-1,x) - a(M-4,x)/256 + 3a(M-5,x)/1024
         + 3a(M-6,x)/4096
         + a(M-4,x-1)/256 - 3a(M-5,x-1)/1024
         - 3a(M-6,x-1)/4096

Q=1,1,1,1,1,1 (e.g., S=AAAAAA)
  a(M,x) = 3a(M-1,x)/4
         + 3a(M-2,x)/16
         + 3a(M-3,x)/64
         + 3a(M-4,x)/256
         + 3a(M-5,x)/1024 + 3a(M-6,x)/4096
         + a(M-1,x-1)/4
         - 3a(M-2,x-1)/16
         - 3a(M-3,x-1)/64
         - 3a(M-4,x-1)/256
         - 3a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096
```

L=7:

Q=0,0,0,0,0,0,1 (e.g., S=ACGTACA)

$$a(M,x) = a(M-1,x) - a(M-7,x)/16384 + a(M-7,x-1)/16384$$

Q=0,0,1,0,0,0,1 (e.g., S=ACGTACG)

$$a(M,x) = a(M-1,x) - a(M-4,x)/256 + a(M-5,x)/256 - a(M-7,x)/16384 + a(M-4,x-1)/256 - a(M-5,x-1)/256 + a(M-7,x-1)/16384$$

Q=0,1,0,0,0,0,1 (e.g., S=ACGTTAC)

$$a(M,x) = a(M-1,x) - a(M-5,x)/1024 + a(M-6,x)/1024 - a(M-7,x)/16384 + a(M-5,x-1)/1024 - a(M-6,x-1)/1024 + a(M-7,x-1)/16384$$

Q=1,0,0,0,0,0,1 (e.g., S=ACGTACA)

$$a(M,x) = a(M-1,x) - a(M-6,x)/4096 + 3a(M-7,x)/16384 + a(M-6,x-1)/4096 - 3a(M-7,x-1)/16384$$

Q=1,0,0,1,0,0,1 (e.g., S=ACGACGA)

$$a(M,x) = a(M-1,x) - a(M-3,x)/64 + a(M-4,x)/64 - a(M-6,x)/4096 + 3a(M-7,x)/16384 + a(M-3,x-1)/64 - a(M-4,x-1)/64 + a(M-6,x-1)/4096 - 3a(M-7,x-1)/16384$$

Q=1,0,1,0,0,0,1 (e.g., S=ACAAACA)

$$a(M,x) = a(M-1,x) - a(M-4,x)/256 + a(M-5,x)/256 - a(M-6,x)/4096 + 3a(M-7,x)/16384 + a(M-4,x-1)/256 - a(M-5,x-1)/256 + a(M-6,x-1)/4096 - 3a(M-7,x-1)/16384$$

Q=1,0,1,0,1,0,1 (e.g., S=ACACACA)

$$a(M,x) = a(M-1,x) - a(M-2,x)/16 + a(M-3,x)/16 - a(M-4,x)/256 + a(M-5,x)/256 - a(M-6,x)/4096 + 3a(M-7,x)/16384 + a(M-2,x-1)/16 - a(M-3,x-1)/16 + a(M-4,x-1)/256 - a(M-5,x-1)/256 + a(M-6,x-1)/4096 - 3a(M-7,x-1)/16384$$

Q=1,1,0,0,0,0,1 (e.g., S=AACCCAA)

$$a(M,x) = a(M-1,x) - a(M-5,x)/1024 + 3a(M-6,x)/4096 + 3a(M-7,x)/16384 + a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096 - 3a(M-7,x-1)/16384$$

Q=1,1,1,0,0,0,1 (e.g., S=AAACAAA)

$$a(M,x) = a(M-1,x) - a(M-4,x)/256 + 3a(M-5,x)/1024 + 3a(M-6,x)/4096 + 3a(M-7,x)/16384 + a(M-4,x-1)/256 - 3a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096 - 3a(M-7,x-1)/16384$$

```
Q=1,1,1,1,1,1,1,1 (e.g., S=AAAAAAA)
  a(M,x) = 3a(M-1,x)/4
         + 3a(M-2,x)/16
         + 3a(M-3,x)/64
         + 3a(M-4,x)/256
         + 3a(M-5,x)/1024 + 3a(M-6,x)/4096
         + 3a(M-7,x)/16384
         + a(M-1,x-1)/4
         - 3a(M-2,x-1)/16
         - 3a(M-3,x-1)/64
         - 3a(M-4,x-1)/256
         - 3a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096
         - 3a(M-7,x-1)/16384
```

L=8:

```
Q=0,0,0,0,0,0,0,1 (e.g., S=ACGTGGTC)
  a(M,x) = a(M-1,x) - a(M-8,x)/65536
         + a(M-8,x-1)/65536

Q=0,0,0,1,0,0,0,1 (e.g., S=ACGTACGT)
  a(M,x) = a(M-1,x) - a(M-4,x)/256 + a(M-5,x)/256
         - a(M-8,x)/65536
         + a(M-4,x-1)/256 - a(M-5,x-1)/256
         + a(M-8,x-1)/65536

Q=0,0,1,0,0,0,0,1 (e.g., S=ACGTTACG)
  a(M,x) = a(M-1,x) - a(M-5,x)/1024 + a(M-6,x)/1024
         - a(M-8,x)/65536
         + a(M-5,x-1)/1024 - a(M-6,x-1)/1024 + a(M-8,x-1)/65536

Q=0,1,0,0,0,0,0,1 (e.g., S=ACGTGTAC)
  a(M,x) = a(M-1,x) - a(M-6,x)/4096 + a(M-7,x)/4096
         - a(M-8,x)/65536
         + a(M-6,x-1)/4096 - a(M-7,x-1)/4096 + a(M-8,x-1)/65536

Q=0,1,0,0,1,0,0,1 (e.g., S=ACGACGAC)
  a(M,x) = a(M-1,x) - a(M-3,x)/64 + a(M-4,x)/64
         - a(M-6,x)/4096 + a(M-7,x)/4096
         - a(M-8,x)/65536
         + a(M-3,x-1)/64 - a(M-4,x-1)/64
         + a(M-6,x-1)/4096 - a(M-7,x-1)/4096
         + a(M-8,x-1)/65536

Q=0,1,0,1,0,1,0,1 (e.g., S=ACACACAC)
  a(M,x) = a(M-1,x) - a(M-2,x)/16 + a(M-3,x)/16
         - a(M-4,x)/256 + a(M-5,x)/256 - a(M-6,x)/4096
         + a(M-7,x)/4096
         - a(M-8,x)/65536
         + a(M-2,x-1)/16 - a(M-3,x-1)/16
         + a(M-4,x-1)/256 - a(M-5,x-1)/256 + a(M-6,x-1)/4096
         - a(M-7,x-1)/4096
         + a(M-8,x-1)/65536

Q=1,0,0,0,0,0,0,1 (e.g., S=ACCCCCCA)
  a(M,x) = a(M-1,x) - a(M-7,x)/16384 + 3a(M-8,x)/65536
         + a(M-7,x-1)/16384 - 3a(M-8,x-1)/65536

Q=1,0,0,1,0,0,0,1 (e.g., S=ACGAACGA)
  a(M,x) = a(M-1,x) - a(M-4,x)/256 + a(M-5,x)/256
         - a(M-7,x)/16384 + 3a(M-8,x)/65536
         + a(M-4,x-1)/256 - a(M-5,x-1)/256
         + a(M-7,x-1)/16384
         - 3a(M-8,x-1)/65536

Q=1,0,1,0,0,0,0,1 (e.g., S=ACAGGACA)
  a(M,x) = a(M-1,x) - a(M-5,x)/1024 + a(M-6,x)/1024
         - a(M-7,x)/16384 + 3a(M-8,x)/65536
         + a(M-5,x-1)/1024 - a(M-6,x-1)/1024
         + a(M-7,x-1)/16384 - 3a(M-8,x-1)/65536

Q=1,1,0,0,0,0,0,1 (e.g., S=AACCCCAA)
  a(M,x) = a(M-1,x) - a(M-6,x)/4096 + 3a(M-7,x)/16384
         + 3a(M-8,x)/65536
         + a(M-6,x-1)/4096 - 3a(M-7,x-1)/16384
         - 3a(M-8,x-1)/65536
```

```
Q=1,1,0,0,1,0,0,1 (e.g., S=AAGAAGAA)
  a(M,x) = a(M-1,x) - a(M-3,x)/64 + a(M-4,x)/64
         - a(M-6,x)/4096
         + 3a(M-7,x)/16384 + 3a(M-8,x)/65536
         + a(M-3,x-1)/64 - a(M-4,x-1)/64
         + a(M-6,x-1)/4096
         - 3a(M-7,x-1)/16384 - 3a(M-8,x-1)/65536

Q=1,1,1,0,0,0,0,1 (e.g., S=AAACGAAA)
  a(M,x) = a(M-1,x) - a(M-5,x)/1024
         + 3a(M-6,x)/4096 + 3a(M-7,x)/16384
         + 3a(M-8,x)/65536
         + a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096
         - 3a(M-7,x-1)/16384 - 3a(M-8,x-1)/65536

Q=1,1,1,1,1,1,1,1 (e.g., S=AAAAAAAA)
  a(M,x) = 3a(M-1,x)/4
         + 3a(M-2,x)/16
         + 3a(M-3,x)/64
         + 3a(M-4,x)/256
         + 3a(M-5,x)/1024 + 3a(M-6,x)/4096
         + 3a(M-7,x)/16384 + 3a(M-8,x)/65536
         + a(M-1,x-1)/4
         - 3a(M-2,x-1)/16
         - 3a(M-3,x-1)/64
         - 3a(M-4,x-1)/256
         - 3a(M-5,x-1)/1024 - 3a(M-6,x-1)/4096
         - 3a(M-7,x-1)/16384
         - 3a(M-8,x-1)/65536
```

TABLE 2: Values of f(x;4,20,Q), E(X), and V(X)
for Q = 1,1,1,1; 0,1,0,1; and 0,0,0,1
(e.g., for Subsequences AAAA, ACAC, and ACGT)

	PROBABILITY		
FREQUENCY	AAAA	ACAC	ACGT
0	0.9499E 00	0.9383E 00	0.9350E 00
1	0.3772E-01	0.5723E-01	0.6366E-01
2	0.9351E-02	0.4156E-02	0.1359E-02
3	0.2288E-02	0.2653E-03	0.9770E-05
4	0.5527E-03	0.1509E-04	0.1629E-07
5	0.1318E-03	0.7665E-06	0.9095E-12
6	0.3099E-04	0.3442E-07	0.
7	0.7190E-05	0.1334E-08	0.
8	0.1643E-05	0.4184E-10	0.
9	0.3698E-06	0.9095E-12	0.
10	0.8175E-07	0.	0.
11	0.1773E-07	0.	0.
12	0.3757E-08	0.	0.
13	0.7749E-09	0.	0.
14	0.1528E-09	0.	0.
15	0.3001E-10	0.	0.
16	0.5457E-11	0.	0.
17	0.9095E-12	0.	0.
E(X)	.06641	.06641	.06641
V(X)	.10506	.07210	.06477

TABLE 3: Example of a Nucleotide Sequence:
Middle repetitive (Alu family) genome fragment - human.
Length 825. From Georgetown University Medical Center's
Nucleic Acid Sequence Database.

POSITION	NUCLEOTIDES
1- 50	CTCGAGGGAGGAGCCCGGGGCTGGGGTACGGAGGCCTCTGCACATCTTAG
51-100	AGTAAAACAAGCAGGAGAGGCTGGGTGCGGTGGCTCATGCCTATAATCCC
101-150	AGCACTTTAGGAGGCTGAGGCGGGCAGATCACCTGAGGTCGGGAGTTCAA
151-200	GACCAGCCTGACCAACAGGGAGAAACCCCATCTTTACTAAAACTACAAAA
201-250	TTAGCTGGGTGTGGTGGCACATGCCTGTAATCCCAGATATTCGGGAGGCT
251-300	GAGGCAGGAGAATCGCTTGAACCTGGGAAGCAGAGGTTGCGCTGAGCCGA
301-350	GATGGCACCATTGCACTCCAGCCTGGGCAACGAGAGCGAAACTCCGTCTC
351-400	AAAAAAACAAAAACAAAAAAATCAAAACAATCAAAAAAACAAGCAGGAGG
401-450	GGCTCTGAGGTGCCTGCAACACCCAGGTACAATCCGTGGCCCTGAGGCCC
451-500	ATCACAGGGAAGGGGTCTTTGCAGCTCTTTCAACCCCCAGCCCAGCATCC
501-550	AAGGAAGCCCAGGGCAGGGAGAAACCTCAGCTGCACCATCAGAGCTCAGA
551-600	ACAGAGAAGGCAGAAATTAGCAGGGAGTGGGGCTGGGGAGGCTTCCTAGA
601-650	AGACGTGTCTCCCGCCTTGCTGGCACTGAGGCCTTGAGGATGGGTCCATA
651-700	CTGGGCCCCCACTGCCAGGGATGCAGATCCGGCCCACTGCTGAAATCTGT
701-750	GCTCCTGGAGCCTCCCTCCTGTTCATGGGCCACAGGCTGTGAAAACCCCA
751-800	GAGTCCTCCCAGGCAGCAAGTTTTGTTTTGTTTTTTGTTTGTTTGCTTGT
801-825	TTGTTTTTTGAGAGTCTGCTCGTCA

TABLE 4: Values of f(x;2,825,Q), E(X), and V(X) for Q = 1,1 and x=30,70 (e.g., for Subsequence CC)

FREQUENCY	PROBABILITY	CUMULATIVE PROBABILITY
30	0.0010	0.0028
31	0.0016	0.0044
32	0.0023	0.0066
33	0.0032	0.0098
34	0.0044	0.0142
35	0.0059	0.0202
36	0.0078	0.0280
37	0.0101	0.0381
38	0.0128	0.0509
39	0.0158	0.0667
40	0.0191	0.0858
41	0.0227	0.1085
42	0.0265	0.1350
43	0.0303	0.1654
44	0.0341	0.1995
45	0.0377	0.2372
46	0.0409	0.2781
47	0.0437	0.3219
48	0.0460	0.3678
49	0.0476	0.4154
50	0.0484	0.4638
51	0.0486	0.5124
52	0.0481	0.5605
53	0.0468	0.6074
54	0.0450	0.6524
55	0.0427	0.6950
56	0.0399	0.7349
57	0.0368	0.7717
58	0.0335	0.8053
59	0.0302	0.8354
60	0.0268	0.8622
61	0.0235	0.8857
62	0.0204	0.9061
63	0.0175	0.9235
64	0.0148	0.9383
65	0.0124	0.9507
66	0.0103	0.9610
67	0.0084	0.9694
68	0.0068	0.9762
69	0.0055	0.9816
70	0.0043	0.9860
E(X)	51.50	
V(X)	67.57	

APPENDIX

Derivation of the Probability Generating Function P(u,v)

Let $P(u,v) = \sum_{M=0}^{\infty} \sum_{x=o}^{\infty} a_{M,x}\, u^M\, v^x$, where $a_{M,x}$ is the probability of x occurrences of a specified string S of length L and overlap capability Q in a string of length M. The approach is to develop a generating function which counts the x occurrences of S as a substring, and to convert the result to a probability generating function. The derivation here is an application of the material in Goulden and Jackson (1983, Section 2.8), restricted to the special case of one distinguished substring, and developed for an alphabet N of n symbols.

Let S be a non null string of length L. A cluster of length M and index t is a string C with a distinguished subset of members $s_{k_1}, s_{k_2}, \ldots, s_{k_t}$ and a distinguished set of substrings $T_1, T_2, \ldots T_t$ with the following properties:

(1) Each T_i is the string S;

(2) The symbol s_{k_i} is the first member of T_i;

(3) The subscript k_1 is 1, and k_t = M-L+1, that is, the first and last L symbols of C are distinguished substrings;

(4) Any consecutive pair of substrings overlap;

(5) Every element of C occurs in at least one T_i.

Note that not every substring of C which is identical with S need be distinguished. For example, let S = ACACAC. Then a cluster of length 12 and index 3 is ACACACACACAC, where the distinguished substrings begin in the first, third, and seventh position. There is a string identical to S which begins in the fifth position, but this is neither distinguished nor counted in the index count.

To introduce the overlap information, the prefix polynomial is used. A prefix of a string S is a non-null string P such that there exist non-null strings X and Y such that S = PX = YP. Let $\{n_1, n_2, \ldots n_u\}$ denote the set of prefix lengths in S. Then the prefix polynomial is $h(x) = \sum_{i=1}^{u} x^{n_i}$. Note that this definition coincides with that given in Section 4.

Note that any cluster C of index t can be uniquely decomposed into an ordered set of t-1 prefixes and a copy of S, with each prefix beginning with a distinguished element and terminating just before the following such element. For the cluster given above, the decomposition is AC,ACAC,ACACAC. Conversely, such an ordered collection gives rise to a unique cluster of index t by reversing the above procedure.

Let $c_{M,t}$ denote the number of clusters of length M and index t relative to the string S. The cluster generating function C(u,v) for S is defined by

$$C(u,v) = \sum_{M=0}^{\infty} \sum_{t=0}^{\infty} c_{M,t}\, u^M v^t.$$

In the following, we use the fact that the generating function for an ordered collection of objects is given by the product of the generating functions for the objects; that is, if A and B are collections of objects with generating functions $\underline{A}$ and $\underline{B}$ respectively, then the collection of objects (a,b) where $a \in A$ and $b \in B$ is $\underline{A}\,\underline{B}$. For further details see Goulden and Jackson (1983).

Lemma. Let S be a string of length L with prefix polynomial h(x). Then the cluster generating function for S is

$$C(u,v) = u^L v/(1-vh(u)).$$

Proof. As noted above, a cluster of index t can be decomposed into an ordered collection of t-1 prefixes and a copy of S. The generating function for $\{S\}$ is $u^L v$, and for the prefixes is vh(u), so the generating function for clusters of weight t is

$$(vh(u))^{t-1}\, u^L v.$$

Summing over all values of t yields

$$C(u,v) = u^L v \sum_{t=1}^{\infty} (vh(u))^{t-1}$$

$$= u^L v/(1-vh(u))$$

as required.

To obtain the generating function for the number of occurrences of S as a substring of all strings of length M from N, it is convenient to work with indexed strings. An indexed string T of length M and index t (relative to a string of length L) is a string of length M with a distinguished subset of entries $s_{k_1}, s_{k_2}, \ldots, s_{k_t}$ and a distinguished set of substrings $T_1, T_2, \ldots, T_t$ with the following properties:

(1) Each T_i is the string S;

(2) s_{k_i} is the first entry of T_i.

Note that unlike clusters, we do not require that the first or last strings of length L be copies of S, nor must every element occur in a distinguished string. Also, adjacent distinguished subsets need not overlap.

Let $d_{M,t}$ denote the number of indexed strings of length M and index t relative to S. Then the index string generating function for S is

$$D(u,v) = \sum_{M=0}^{\infty} \sum_{t=0}^{\infty} d_{M,t}\, u^M\, v^t.$$

Lemma. Let S be a string of length L with cluster generating function C(u,v). Then the indexed string generating function for S is

$$D(u,v) = (1-nu-C(u,v))^{-1}$$

where n is the number of alphabet symbols.

As with clusters, indexed sets can be uniquely decomposed into ordered collections; this time the entries will either be a single element from N or a cluster. To obtain the decomposition, work from the beginning to the end,

examining each character and treating it as a single entry in the ordered collection until a distinguished element s_{k_i} is hit. This will be a beginning of a unique cluster, which is then used as an entry in the collection. The scan continues until the end is reached. Conversely, any ordered collection of single elements and clusters gives rise to a unique corresponding indexed sequence. Since there are n alphabet symbols, the generating function for the set of entries for each position in the collection is nu+C(u,v), and the generating function for all collections of length w is $(nu+C(u,v))^w$. Adding over all w yields the result.

Note that the introduction of symbols between clusters in the creation of indexed sets can introduce extra copies of S (which are not distinguished in the cluster). Also, there may be undistinguished copies of S within the clusters themselves. Let T be a string of length M from N which contains precisely k substrings identical to S. By considering the construction of indexed strings, we see that T is counted in D(x,y) precisely once for each subset of the k copies of S. That is, D(x,y) is an "at least" generating function for the set of strings counted by the number of copies of S which it contains, in the sense of the principle of inclusion and exclusion (see, for example, Goulden and Jackson (1983)).

In generating function form, the principle of inclusion and exclusion states that if f(z) is the generating function for the number of objects which contain "at least" k properties (in the above sense), then the generating function g(z) for the number of objects with exactly k properties is given by g(z) = f(z-1). Therefore if $f_{M,x}$ denotes the number of strings of length M from N which contain precisely x substrings identical to S, and

$$F(u,v) = \sum_{M=0}^{\infty} \sum_{x=0}^{\infty} f_{M,x}\, u^M\, v^x,$$

then

$$F(u,v) = (1-nu-C(u,v-1))^{-1}.$$

There are n^M possible sequences of length M, so to obtain the probability generating function P(u,v), replace u by u/n in F(u,v). In particular, if n = 4, then

$$P(u,v) = \frac{1-(v-1)h(u/4)}{[1-(v-1)h(u/4)](1-u)-(u/4)^L(v-1)}$$

as in Section 4.

ANALYTICAL STUDIES BRANCH

RESEARCH PAPER SERIES

No.

1. BEHAVIOURAL RESPONSE IN THE CONTEXT OF SOCIO-ECONOMIC MICROANALYTIC SIMULATION by Lars Osberg

2. UNEMPLOYMENT AND TRAINING by Garnett Picot

3. HOMEMAKER PENSIONS AND LIFETIME REDISTRIBUTION by Michael Wolfson

4. MODELLING THE LIFETIME EMPLOYMENT PATTERNS OF CANADIANS by Garnett Picot

5. JOB LOSS AND LABOUR MARKET ADJUSTMENT IN THE CANADIAN ECONOMY by Garnett Picot and Ted Wannell

6. A SYSTEM OF HEALTH STATISTICS: TOWARD A NEW CONCEPTUAL FRAMEWORK FOR INTEGRATING HEALTH DATA by Michael Wolfson

7. A PROTOTYPE MICRO-MACRO LINK FOR THE CANADIAN HOUSEHOLD SECTOR SESSION 3, MACRO/MICRO LINKAGES - HOUSEHOLDS by Hans Adler and Michael Wolfson

8. NOTES ON CORPORATE CONCENTRATION AND CANADA'S INCOME TAX by Michael Wolfson

9. THE EXPANDING MIDDLE: SOME CANADIAN EVIDENCE ON THE DESKILLING DEBATE by John Myles

10. THE RISE OF THE CONGLOMERATE ECONOMY by Jorge Niosi

11. ENERGY ANALYSIS OF CANADIAN EXTERNAL TRADE: 1971 and 1976 by K.E. Hamilton

12. NET AND GROSS RATES OF LAND CONCENTRATION by Ray Bollman and Philip Ehrensaft

13. CAUSE-DELETED LIFE TABLES FOR CANADA (1921 to 1981): An Approach Towards Analysing Epidemiologic Transition by Drhuva Nagnur and Michael Nagrodski

14. THE DISTRIBUTION OF THE FREQUENCY OF OCCURRENCE OF NUCLEOTIDE SUBSEQUENCES BASED ON THEIR OVERLAP CAPABILITY by Jane F. Gentleman and Ronald C. Mullin

15. IMMIGRATION AND THE ETHNOLINGUISTIC CHARACTER OF CANADA AND QUEBEC by Réjean Lachapelle

16. INTEGRATION OF CANADIAN FARM AND OFF-FARM MARKETS AND THE OFF-FARM WORK OF WOMEN, MEN AND CHILDREN by Ray Bollman and Pamela Smith

For further information, contact the Chairperson, Publication Review Committee, Analytical Studies Branch, R.H. Coats Bldg., 24th Floor, Statistics Canada, Tunney's Pasture, Ottawa, Ontario K1A 0T6.